

Quotatool: The Sysadmin's Daily Chores

Part Taker

A disk quota restricts the hard disk space on a (file) server for a user or group. There are two distinct limits: the soft limit that may be exceeded for a certain time period, and the hard limit that may not be exceeded under any circumstances. When a user hits this limit, the system will refuse to perform write operations in the restricted area, instead presenting the user with a *quota exceeded* message. The period during which overusage is tolerated is referred to as the grace period.

You must first ensure that your kernel supports quotas, if you want to activate them, i.e. the kernel must be compiled with the *quota-support: yes* flag set. In addition, you will need to set the *usrquota* or *grpquota* parameter for the corresponding mountpoint in */etc/fstab*, for example:

```
/dev/hda4 /home ext3 2
defaults,usrquota,grpquota 1 1
```

The following tools are used for quota administration: *quota*, *quotaon*, *quotaoff*, *quotacheck*, *repquota* and *edquota*. Too complicated? Sure, but luckily there is an easy way to go: Quotatool[1]. Quotatool Version 1.2.1 is currently available for Linux, Solaris and AIX. A tarball of the tool is available and after expanding it, compilation should be child's play, if you just keep to the standard: *./configure; make; make install*.

And this is how you set a quota for the user *hugo*'s home directory:

```
quotatool -u hugo -b -q 50M -l 70M /home
```

The *-b* parameter indicates a block limit. You can replace *-b* with *-i* to set an inode limit. The *-q 50M* entry sets the soft limit to 50 Mbytes, and *-l 70M* sets the hard limit to 70 Mbytes. Finally, you need to indicate the mountpoint that will be the

root for this limit. You can apply quotas to complete partitions (for example */dev/hda4*), or as an alternative refer to the corresponding mountpoint, that is, */home* in this case.

BY CHARLY KÜHNAST

Setting Group Quotas

Follow the same pattern to set group quotas. You can also use the following syntax to apply the same limits defined in our previous example to the members of group *users*:

```
quotatool -g users -b -q 50M -l 70M /home
```

And this is how you define the grace period:

```
quotatool -u -b -t "1 week" 2 /home
```

The parameter *-t "1 week"* is critical. The following alternative units of time are available: *sec*, *min*, *hour*, *day*, and *month*.

You can only set two grace periods per type (user quota and group quota): one for the block limit and one for the inodes. So you will not be able to define different grace periods for two members of a group whose home directories are in the same partition. And that is why you do not need a user name for the parameter *-u* in the previous example – the time limits you set, apply to everyone.

However, the grace period can be extended on request. Imagine that the user Hugo needs an extension of the soft

limit, because the limit of one week is too short. After bribing the Admin (with a large drink – Admins are only human after all), Hugo might ask his resident connoisseur to type the following

```
quotatool -u hugo -b -r /home
```

and thus extend the grace period. This does not mean a week's extension, but simply resets the grace period. The user will be happy and the Admin can take care of more important business – like finishing off that drink, for example. ■



INFO

[1] <http://devel.duluo.net/quotatool>

THE AUTHOR

Charly Kühnast is a Unix System Manager at the datacenter in Moers, near Germany's famous River Rhine. His tasks include ensuring firewall security and availability and taking care of the DMZ (demilitarized zone). Although Charly started out on IBM mainframes, he has been working predominantly with Linux since 1995.

